

Implementasi Monitoring Jaringan Berbasis Web Terintegrasi Cacti Menggunakan Notifikasi Telegram

Manda Sari¹, Titania Halim Putri², Carly Dosen³, M. Diarmansyah Batubara⁴

¹Fakultas Teknologi dan Ilmu Komputer

^{2,3}Program Studi Sistem Informasi Universitas Prima Indonesia

¹desumanda@gmail.com, ²titaniahalimputri@gmail.com, ³carlydosen@gmail.com,

⁴diarmansyahbatubara@unprimdn.ac.id

Abstract

Improving the quality of services on the network must be implemented and run easily in every institution, school, home, or other place that uses a computer network as an important means. One appropriate way is to find out network disruptions that occur in a fast time period, so that the interference can be overcome without needing a long time. Knowing the disruption that occurs in a faster time period is not easy if done the old way, that is network users realize that the network is not connected and report it to the network technician. After receiving the report the damaged network will be repaired. With a web-based system that is integrated with cacti, it can monitor and provide information via telegram messages in real time. The technician will immediately fix the disruption that occurred before the report from the user, so that network disruptions that occur will not last long.

Keywords : *cacti, web, telegram, networking*

1. PENDAHULUAN

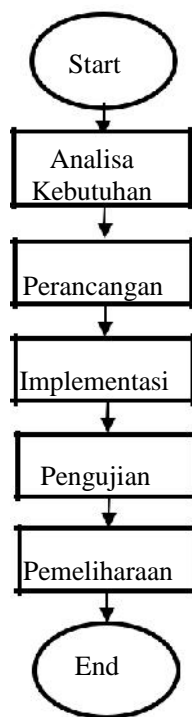
Jaringan komputer saat ini terus semakin berkembang karena itulah jaringan komputer juga harus diiringi dengan meningkatnya kualitas layanan pada jaringan komputer. Pelayanan yang baik kepada pengguna di dalam jaringan komputer haruslah dapat difasilitasi dengan maksimal sehingga tidak mengganggu komunikasi jaringan komputer yang ada. Namun yang masih sering terjadi pada layanan jaringan adalah ketika layanan pengguna mati maka pengguna harus menyampaikan pengaduannya secara manual melalui telepon agar teknisi melakukan kunjungan ke lokasi untuk perbaikan jaringan. Hal ini tentu tidak efektif, sehingga dengan adanya sistem ini diharapkan pada suatu tempat yang menyediakan jaringan kepada pengguna dapat lebih cepat dalam melakukan perbaikan terhadap gangguan yang terjadi.

Pada penelitian sebelumnya, cacti hanya digunakan sebagai monitoring server, *bandwidth* dan perangkat pengguna. Peneliti mengembangkan adanya monitoring jaringan berbasis web yang terhubung dengan *cacti*, sehingga web dan *cacti* saling berhubungan untuk memonitor suatu perangkat. Jika pada *cacti* atau web perangkat termonitor *down*, maka akan muncul pemberitahuan pada telegram teknisi atau orang yang melakukan perbaikan dalam rentang waktu 5 menit sekali yang berisi informasi perangkat yang *down* dan berapa lama perangkat tersebut *down*.

Sehingga, jaringan yang *down* dapat termonitor secara *realtime* dan perbaikan pada jaringan dilakukan sedini mungkin. Teknisi atau orang yang melakukan perbaikan juga dapat memantau jaringan secara *portable* karena sifat telegram yang dapat diakses dari dan dimana saja. Pada implementasi ini *cacti* juga berfungsi sebagai database pada web.

2. METODE PENELITIAN

Pengembangan penelitian ini dikembangkan dengan metode *waterfall*, karena penggunaan metode *waterfall* akan membantu sistem menjadi lebih mudah dipahami karena sifat metode *waterfall* yang sistematis dan berurutan untuk membangun sebuah sistem. Tahapan dari metode *waterfall* sebagai berikut.



Gambar 1. Metode *waterfall*

Pada tahap analisis kebutuhan peneliti akan menganalisis perangkat lunak dan perangkat keras yang dibutuhkan untuk membangun sistem, kemudian pada tahap perancangan akan memberikan gambaran tentang desain yang akan ditampilkan pada sistem. Pada tahap implementasi sistem akan dilengkapi pengkodean (*coding*) pada sistem ke dalam bentuk perintah-perintah yang dimengerti komputer dengan bahasa pemrograman, dan pada tahap pengujian pada sistem untuk memastikan sistem dapat berjalan sesuai dengan perancangan desain dan semua fungsi pada perancangan desain dapat dipergunakan dengan baik tanpa ada kesalahan. Kemudian, sistem akan dapat diimplementasikan. Tahap terakhir adalah pemeliharaan dimana dalam tahap ini akan mengoreksi *error* pada sistem, dan untuk meningkatkan sistem menjadi lebih baik pemeliharaan dapat dilakukan oleh *administrator* atau *programmer*.

Sistem yang dibangun merupakan rancang bangun pengembangan *cacti*, dimana adanya fitur notifikasi melalui telegram dan terhubung ke web. Pada perancangan ini web adalah kumpulan perintah-perintah linux (*script base command*) yang akan memanggil data masukan (*input*) dari basis data *cacti* sehingga menghasilkan *output* berupa *file text* yang nantinya akan dikirimkan oleh telegram sebagai pesan notifikasi gangguan. web ini nantinya juga akan memberikan informasi gangguan setiap lima menit.

Cacti juga merupakan *tools* yang berguna sebagai sumber data pada web, memberi *inputan* berupa *list host down*, dan sistem *monitoring* jaringan untuk memantau aktivitas jaringan yang memberikan peringatan bila terjadi masalah pada jaringan dan informasi durasi jaringan *down*.

Telegram merupakan media *chatting* yang dapat digunakan dalam mode *command line interface* atau terminal yang digunakan sebagai wadah informasi pemberitahuan gangguan.

3. HASIL DAN PEMBAHASAN

Untuk hasil perancangan yang telah disusun dapat berjalan dengan baik, sehingga pada penelitian ini harus dilengkapi dengan *software* dan *hardware* yang menunjang sistem. Dalam penelitian ini dibutuhkan *software* dan *hardware* sebagai berikut:

1. Laptop Acer Core i3

2. Router TP-Link TL-WR840N-V6
3. Laptop Lenovo Core i3
4. Personal Komputer
5. Kabel Ethernet LAN RJ-45
6. Router ZTE
7. Sistem Operasi Linux
8. *PhpMyAdmin* sebagai *database* server dan web server
9. *Scripting Language PHP7,TCL,Shell Script* , ini adalah bahasa pemrograman yang digunakan pada sistem.
10. *Telegram-Cli for Linux* sebagai wadah untuk mengirim notifikasi gangguan ke pengguna
11. *Cacti* dan *software* pendukung yaitu *RRD Tool* sebagai *software monitoring* yang akan diinstal pada komputer server

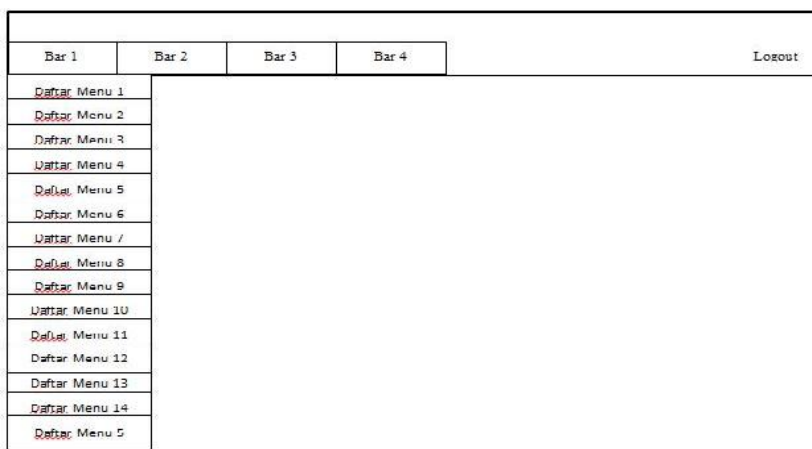
Sebelum melakukan implementasi maka peneliti melakukan perancangan desain sistem agar sistem pada tahap pengujian berjalan sesuai dengan yang diharapkan.

Gambar 2. Login *cacti*

Pada Gambar 2. *Login Cacti* berupa gambaran tampilan saat *administrator* akan melakukan proses memasukkan *username* dan *password* untuk dapat mengakses *cacti*.

Gambar 3. Menu utama pada *cacti*

Pada Gambar 3. Menu Utama Pada *Cacti* berupa gambaran tampilan saat *administrator* telah melakukan proses memasukkan *username* dan *password* pada *cacti*.



Gambar 4. Menu utama pada website

Pada Gambar 4. Menu Utama Pada Website merupakan perancangan halaman utama pada website yang berfungsi untuk memonitor jumlah perangkat yang berstatus *up* atau *down*.

Gambar 5. Menu data *input* pada website

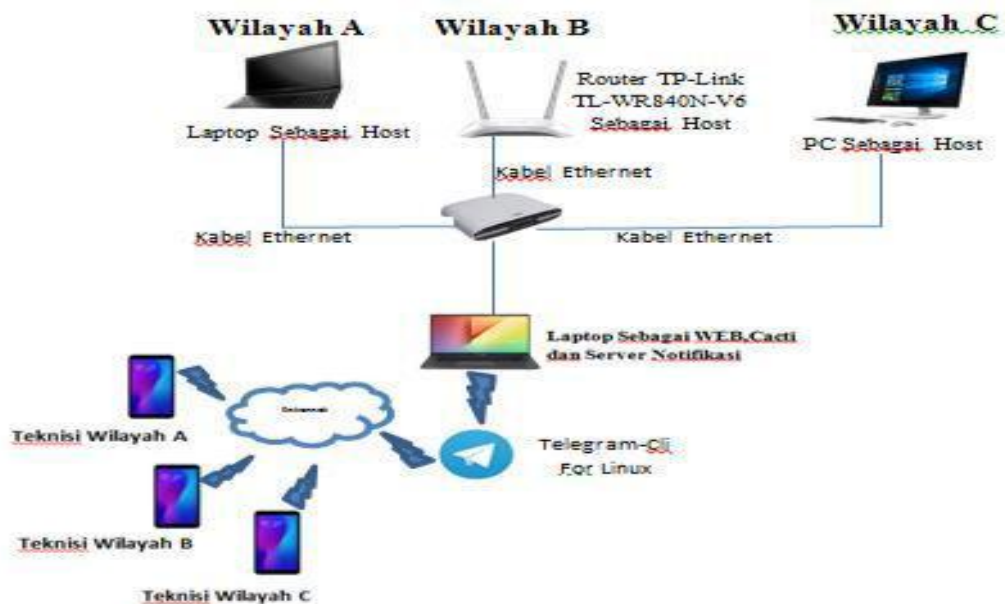
Pada Gambar 5. Menu Data *Input* Pada Website merupakan perancangan halaman untuk menambahkan perangkat baru yang akan di monitor. Nantinya, jika perangkat baru di tambahkan melalui halaman ini, secara otomatis akan masuk pada monitoring *cacti*.

Setelah perancangan selesai , maka web di lengkapi dengan kode pemograman yang pada akhirnya akan menghasilkan informasi pada telegram dengan rentang waktu yang di tentukan, pada pengujian ini akan di lakukan dengan rentang waktu 5 menit.

Pengujian akan menggambarkan kondisi yang terjadi jika *server* di jalankan. Pengujian akan dilakukan dengan menguji web dan hasil mengirimkan notifikasi gangguan melalui telegram, apakah sudah sesuai dengan yang sudah di rancang.

Perangkat yang di pakai sebagai *host* untuk uji coba merupakan perangkat keras yang dapat mewakili pada wilayah yang sudah di tentukan. Sebagai contoh, Router TP-Link wilayah A dianalogikan sebagai *server*. Jadi, jika terdeteksi terjadinya gangguan pada server A maka secara otomatis akan muncul pemberitahuan gangguan yang di kirimkan ke grup telegram Teknisi A. Untuk wilayah B dan wilayah C cara kerja dan analogi nya sama seperti Wilayah A. Perangkat tersebut dirasa mampu untuk mewakili masing-masing wilayah.

Berdasarkan penjelasan tersebut, maka dapat disusun arsitektur pengujian untuk membuktikan apakah Web yang di bangun memiliki hasil yang sesuai dengan yang dirancang atau tidak. Arsitektur pengujian terdapat pada Gambar 6.



Gambar 6. Arsitektur pengujian

Dari Gambar 6, dijelaskan sebuah laptop akan difungsikan sebagai *Cacti* dan Web. Router TP-link dipakai sebagai *host* yang akan di uji masing-masing wilayahnya. *Mobile* wilayah A, *Mobile* wilayah B, *Mobile* wilayah C dipakai sebagai penerima notifikasi gangguan jaringan berdasarkan wilayah operasinya.

Konfigurasi sehingga *cacti* dapat terkoneksi dengan telegram sesuai wilayahnya adalah pada penelitian ini web menggunakan *database cacti*, sehingga *database cacti* dan web adalah sama, *host* yang di *input* pada web akan muncul pada *cacti*. Tampilan pada *cacti* berguna sebagai monitoring. *Cacti* tidak langsung terhubung dengan telegram, *cacti* hanya terhubung *database* dengan web.

Program web dan telegram berada di dalam satu server. Pada server dibuat direktori untuk folder telegram, pada penelitian ini saya membuat folder telegram pada direktori `/home/skripsi/alert_tg` lalu menggunakan teks editor nano/vim untuk membuat file `send_medan`, `send_lubukpakam`, `send_binjai`.

Pada isian masing-masing file, adalah perintah untuk mengaktifkan telegram-cli dengan `/var/lib/snapd/snap/bin/telegram-cli -W -e` lalu menggunakan perintah telegram-cli untuk mengirimkan file teks yang telah dibuat oleh server yaitu `send_text <peer> <file>`. `<peer>` Nama Grup Telegram/Id Telegram Tujuan `<file>` Lokasi file yang ingin dikirimkan. Sehingga web, selain untuk memonitor juga merupakan jembatan agar *cacti* dapat di monitor secara *portable* menggunakan telegram.

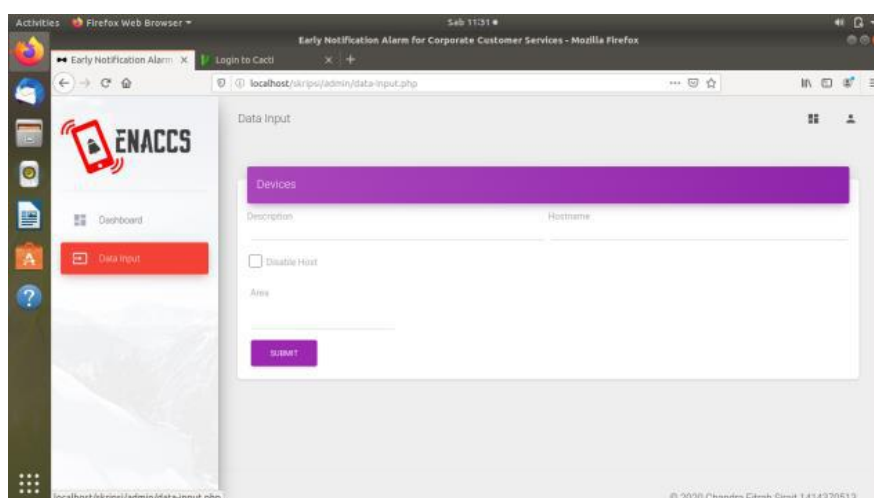
Proses *login* pada *cacti*, dimana *network administrator* akan mempunyai akses penuh terhadap *cacti*. Jika *login* pada *cacti* berhasil maka *network administrator* dapat melakukan *input device* pada *cacti*.

Tabel 1. Proses Login Pada Cacti

Input	Proses	Output
Login Pada Cacti	Masukkan Username dan Password	Login Cacti Berhasil



Gambar 7. Halaman login cacti



Gambar 8. Halaman menu utama cacti

Pengujian *Host* pada *cacti* dilakukan agar *host* dapat di deteksi dan dikirimkan, hal pertama yang harus dilakukan adalah mendaftarkan *host* kedalam *cacti*. Ketika *host* telah di daftarkan melalui menu input pada website, secara otomatis *host* akan muncul pada Cacti. Berikut pada Gambar 9, adalah menu *input* pada web.

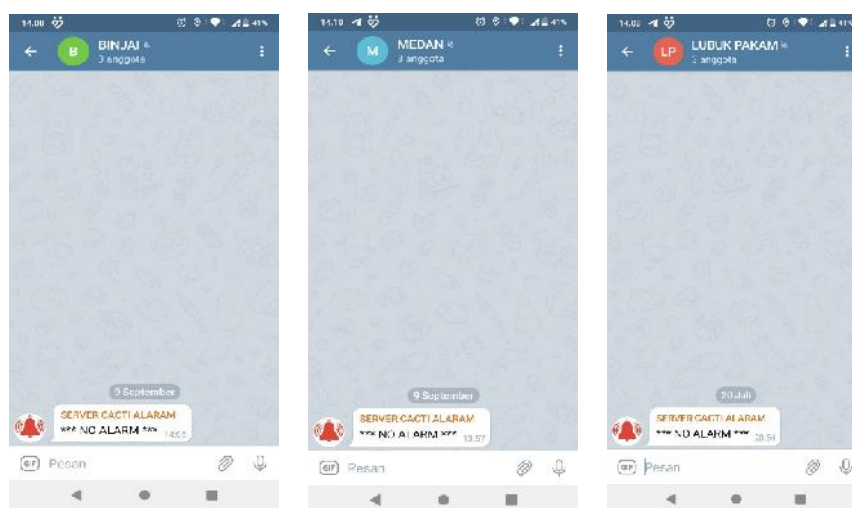
Tabel 2. Proses Input Pada Web

Input	Proses	Output
Mendaftarkan Hostname Pada Web	Masukkan Deskripsi, Hostname dan Kode wilayah	Data Berhasil Di Input

Tabel 4. Pengujian Manajemen Grup Telegram Berdasarkan Wilayah

Input	Proses	Output
Membuat grup telegram dengan penamaan grup sesuai wilayahnya	Grup terdaftar dan tampil di aplikasi telegram	Grup telegram tampil di aplikasi telegram dan akun telegram server berhasil masuk dan mengirimkan pesan "NO ALARM" ke semua grup telegram
Invite akun telegram ke semua grup telegram	Akun telegram server berhasil masuk ke semua grup telegram	

Untuk mengetahui hasil pengujian manajemen grup telegram berdasarkan wilayah dapat dilihat pada Gambar 11.

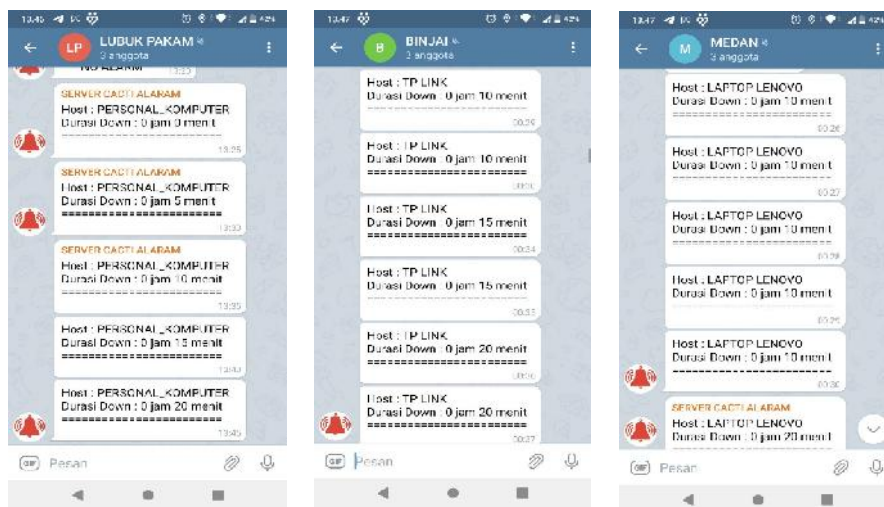


Gambar 11. Hasil pengujian manajemen grup telegram

Tabel 5. Pengujian Pengiriman Notifikasi Ketika Terdapat *Host Down*

Input	Proses	Output
Cabut Kabel power host yang terdaftar	Tunggu 5 menit maka server akan otomatis menjalankan perintah	Notifikasi gangguan terkirim dan diterima pada grup telegram sesuai wilayahnya setiap 5 menit sekali

Hasil pada pengujian pengiriman notifikasi gangguan ketika terdapat *host down* adalah seperti yang tertera pada Gambar 12.



Gambar 12. Pengujian pengiriman notifikasi ketika terdapat *host down* berdasarkan wilayah

Tabel 6. Pengujian Pengiriman Notifikasi Ketika Tidak Terdapat *Host Down*

Input	Proses	Output
Pastikan semua host UP pada Cacti	Cek Kondisi host pada menu device cacti	Notifikasi "No Alarm" terkirim dan diterima di semua grup telegram

Hasil pengujian pengiriman notifikasi gangguan ketika tidak terdapat *host down* adalah seperti yang tertera pada Gambar 13 berikut.



Gambar 13. Hasil pada pengujian pengiriman notifikasi jika tidak terdapat *host down* berdasarkan wilayah

4. KESIMPULAN

Setelah dilakukan pengujian terhadap sistem, maka dapat ditarik kesimpulan berkenaan dengan hasil pengujian, yaitu :

1. Saat tidak terjadi gangguan pada *host* maka notifikasi berupa pesan "NO ALARM" terkirim ke grup Telegram masing-masing wilayah.

2. Notifikasi gangguan pada telegram terkirim secara otomatis setiap lima menit sekali tanpa harus *me-refresh* setiap saat.
3. Web belum mampu untuk melakukan *monitoring* secara *real time*, jadwal rentang monitor minimal hanya 5 menit sekali dan pengiriman notifikasi gangguan hanya di kirim 5 menit sekali.
4. Sehingga web, selain untuk memonitor juga merupakan jembatan agar *cacti* dapat di monitor secara *portable* menggunakan telegram.

Web yang dirancang adalah fitur baru yang diintegrasikan dengan *cacti* untuk mengatasi kekurangan pada *cacti*, salah satu kekurangan pada *cacti* adalah *cacti* tidak mampu diakses dimana saja. Dengan adanya web yang diintegrasikan ke *cacti* ini akan menyampaikan notifikasi dalam bentuk pesan melalui aplikasi telegram yang dapat diakses dari dan dimana saja. Sehingga, monitor jaringan dapat dilakukan dengan mudah dengan pesan yang otomatis terkirim dengan rentang waktu yang ditentukan.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Universitas Prima Indonesia dan semua pihak yang telah memberikan waktu, ilmu, arahan, serta dukungan terhadap penelitian ini.

BAHAN REFERENSI

- [1] Asmunin, A., & Khamdani, W. 2016 . Sistem Monitoring Resource pada Jaringan FMIPA Unesa dengan Protocol SNMP. *MULTINETICS*, 2(1), 8-12.
- [2] Dwitama, I. A. Monitoring Perangkat Keras Jaringan Berbasis Cacti Menggunakan Voice Alert Dan Wake On Lan (Studi Kasus Jaringan Prodi Teknik Informatika Upn “Veteran” Jatim).
- [3] Kurniadi, W., & Jusmin, V. B. 2017 . Aplikasi Monitoring Jaringan Menggunakan Plugin Cacti Melalui Website Pada Mtsn Model Palopo. *Prosiding Semantik*, 1(2), 247.
- [4] Implementasi Monitoring Jaringan Menggunakan Cacti Dan Web Authentication Menggunakan Kerberos Pada Man 1 Bojonegoro. Issn : 2442-5826 E-Proceeding Of Applied Science : Vol.1, No.3 Desember 2015 | Page 2323.
- [5] Rinaldo, R. 2016 . Implementasi Sistem Monitoring Jaringan Menggunakan Mikrotik Router OS di Universitas Islam Batik Surakarta. Emitor: Jurnal Teknik Elektro, 16(02), 5-12.
- [6] Udjulawa, D. 2019 . Rancang Bangun Network Monitoring Dan Bandwidth Monitoring Dengan Menggunakan Aplikasi Cacti Pada PT. XYZ. *JATISI (Jurnal Teknik Informatika dan Sistem Informasi)*, 6(1), 1-17.
- [7] Bayunadi, I., Rochim, A. F., & Satoto, K. I. 2013 . Network Monitoring Service Berbasis Simple Network Management Protocol Menggunakan Aplikasi Cacti. *Transmisi*, 15(4), 191-198.
- [8] Bayunadi, I., Rochim, A. F., & Satoto, K. I. 2013. Network Monitoring Service Berbasis Simple Network Management Protocol Menggunakan Aplikasi Cacti. *Transmisi*, 15(4), 191-198..
- [9] Rifai, Bakhtiar; Nuryadi, Nanang; Ripai, Amarulloh. IMPLEMENTASI TELEGRAM NOTIFICATION ALERT PADA NETWORK MONITORING SYSTEM DENGAN NAGIOS. *Jurnal Mantik Penusa*, 2019, 3.3.
- [10] Ikhsanto, Muhammad Nur; Nugroho, Handoyo Widi. analisis performa dan desain jaringan komputer menggunakan top-down network desain studi kasus pada CV. Merah Putih. *Jurnal Teknologi Informasi Magister*, 2016, 1.01: 69-82.
- [11] Nasution, Muhammad Irwan Padli., Samsudin, S, 2018. Using google location APIs to find an accurate criminal accident location. *International Journal of Engineering & Technology*, 7(3), 2018, p1818-1820. <https://doi.org/10.14419/ijet.v7i3.14144>
- [12] Fadhila Nisya Tanjung, Muhammad Irwan Padli Nasution, 2012, "Implementasi Pemrograman Java Untuk Alert Intrusion Detection System", pematang siantar, 31 agustus – 2 september 2012, ISBN 978-602-18749-0-5, <https://www.researchgate.net/publication/307973619>
- [13] Nasution, Muhammad Irwan Padli, 2008, Urgensi Keamanan Pada Sistem Informasi, Jurnal Iqra' Volume 02 Nomor 02.